21

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/401,251	09/23/1999	CATHERINE M. KEENE	A0653-1160	4434

7590 07/15/2003

DAVID R. STEVENS
STEVENS & WESTBERG, LLP
99 NORTH FIRST STREET
SUITE 201
SAN JOSE, CA 95113

EXAMINER	
PHAM, HUNG Q	
ART UNIT	PAPER NUMBER
2172	13

DATE MAILED: 07/15/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Paper No. 13

Application Number: 09/401,251
Filing Date: September 23, 1999
Appellant(s): KEENE ET AL.

David R. Stevens Reg. No. 38,626
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 04/22/2003.

(1) Real Party in Interest

A statement identifying the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

A statement identifying the related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

(3) Status of Claims

The statement of the status of the claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Invention

The summary of invention contained in the brief is correct.

(6) Issues

The appellant's statement of the issues in the brief is correct.

(7) Grouping of Claims

Appellant's brief includes a statement that claims 1, 6-7, 10-12 of group I, claims 2-5, 8 of group II, claim 9 of group III, claims 13-15 of group IV, claim 16 of group V do not stand or fall together and provides reasons as set forth in 37 CFR 1.192(c)(7) and (c)(8).

(8) Claims Appealed

The copy of the appealed claims contained in the Appendix to the brief is correct.

(9) Prior Art of Record

6,052,688	Thorsen	04-2000

(10) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

The changes made to 35 U.S.C. 102(e) by the American Inventors Protection Act of 1999 (AIPA) do not apply to the examination of this application as the application being examined was not (1) filed on or after November 29, 2000, or (2) voluntarily published under 35 U.S.C. 122(b). Therefore, this application is examined under 35 U.S.C. 102(e) prior to the amendment by the AIPA (pre-AIPA 35 U.S.C. 102(e)).

Claims 13-16 are rejected under 35 U.S.C. 102(e) as being anticipated by Thorsen [USP 6,052,688].

Regarding to claim 13, Thorson teaches a computer program product for use with a computer system, ***a central processing unit and means coupled to the central processing unit for storing a database to automatically manage objects for viewing and marking an object having varying formats without the use of any originating application of a file to view the object*** (FIG. 3B; Col. 5, line 63-Col. 6, line 8), comprises: ***computer readable code means for establishing an object in a storage location*** (FIG. 2, col. 8); ***computer readable code means for identifying a user to have limited access to information associated with the object*** (Col. 11, lines 6-65); ***computer readable code means for establishing privilege access criteria that define the scope of access of a version of the object***

for the user (Col. 4, lines 30-33; Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 22-33); ***computer readable code means for receiving an object request by a requestor*** (Col. 11, lines 56-65); ***computer readable code means for verifying the requestor's user privilege access criteria*** (Col. 11, lines 49-60); and ***computer readable code means for transmitting a version of the requested object in the form of a redacted document that marks information according to the requestor's user privilege access criteria*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 22-33; Col. 11, line 60-Col. 12, line 67).

Regarding to claim 14, Thorsen teaches a computer program product for using with a data processing and storage system for obtaining a view of a database or a subset of a database (Col. 20, lines 38-61) and controlling read and write operations by using access control parameters (Col. 15, lines 18-21). The Thorsen computer program product comprises: ***identifying a user to have access to the object*** (Col. 11, lines 6-60); ***establishing privilege access criteria that define the scope of access of a version of the object for the user*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 27-33); ***receiving an object request by a requestor*** (Col. 11, lines 56-65); ***verifying the requestor's user privilege access criteria*** (Col. 11, lines 49-60); ***transmitting a redacted version of a requested object in the form of a document file containing the version of the requested object that was filtered according to the requestor's user privilege access criteria*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 22-33; Col. 11, line 60-Col. 12, line 67).

Regarding to claim 15, Thorsen teaches a computer server having a data base for storing data pertaining to product information, a method of securely transferring data between a source and an access destination (Abstract and FIG. 3B) comprises:

establishing an object in a storage location (FIG. 2, col. 8); ***identifying a user to have limited access to the object*** (Col. 11, lines 6-60); ***establishing privilege access criteria that define the scope of access of a version of the object for the user*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 27-33); ***receiving an object request by a requestor*** (Col. 11, lines 56-65); ***verifying the requestor's user privilege access criteria*** (Col. 11, lines 49-60); ***setting up a version of an object and associated documents according to user access privileges for transmission to the user***; and ***transmitting a redacted version of the requested object that set up according to the requestor's user privilege access criteria, wherein the access criteria defines the information in which a user has privileges of access to the version of the requested object*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 22-33; Col. 11, line 60-Col. 12, line 67).

Regarding to claim 16, Thorsen teaches a system for controlling access to and associating data in an application independent fashion, which enables data of different nature to be handled in a conform way by using data access node (Col. 3, lines 9-22) as ***an application server*** having access to a database for storing objects and associated documents, a method of securely transferring a version of an object and associated documents from the application server to a user system via a network (FIG. 3B, Col. 4, lines 16-38) comprises: ***establishing privilege access criteria that define the scope of access permitted to a user of a version of the object that may be set up and sent to the privileged user*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 27-33); ***receiving an object request by a user via a network for access to a version of an object to which the user has access privileges*** (Col. 11, lines 56-65); ***verifying the requestor's user privilege access criteria*** (Col. 11, lines

49-60); ***setting up a version of an object and associated documents according to user access privileges for transmission to the user***; and ***transmitting a version of the requested object that was set up according to the requestor's user privilege access criteria in the form of a document file that includes a version of the requested object and a version of associated documents via the network*** (Col. 10, line 35-Col. 11, line 5 and Col. 11, lines 22-33; Col. 11, line 60-Col. 12 line 67).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 1-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Thorsen [USP 6,052,688].

Regarding to claim 1, Thorsen teaches a system for controlling access to and associating data in an application independent fashion. The Thorsen system enables the latest version of data of different nature to be handled in a conform way and for allowing different views of stored data objects depending on different aspects of the stored objects or different access rights of a user (Col. 3, lines 9-22; Col. 4, lines 39-56). The Thorsen system comprises: ***a database for storing an object and associated information, the object comprising distinguishable groups of data, each group or data having associated access criteria for access to the groups of data*** (FIG. 3B, Col. 8, lines 43-56; Col. 10, line 42-Col. 11, line 26). As shown in FIG. 3B, Thorsen discloses a state of the art database 52, storing data in a number of tables 54 and being provided with specific data associations. Depending on a control file, the data associations are rearranged; references or pointers to each of the selected data items of the tables 54 are arranged and stored in a number of data access nodes 56 as ***an application server***. A client 58, for example an application program, communicates with the access nodes 56 of the access structure and a new interface is provided between the old database and the user (Col. 8, lines 42-52). This indicates ***an application server configured to control access to data stored in the database***. As shown in FIG. 8 is a flow chart for implementing an object access control means or an object filter. The object filter is used in conjunction with the access control to protect an object referred to by an access node. In step 124, upon a user command (cmd=dir) all objects are copied to or listed in an object list 128.

Thereafter, in step 132 every object in the object list is checked in respect of whether or not the user is permitted access to it. An object access list 134 is thereby used as a check reference. If access right exists for an object, a copy of or a reference to that object is input in a user list, step 136, and is communicated to the user in step 138 by call 140 (Col. 4, lines 30-33; Col. 10, lines 47-60; Col. 11, line 66-Col. 12, line 11). This indicates ***the application server configured to set up and send a document file having a representation of an object and associated documents that are stored in the database.*** Thorsen further disclosed ***a memory for storing software code for controlling the operation of the application server*** (FIG. 1, Col. 7, lines 5-16 and Col. 21, lines 25-51) and ***access application code stored in the memory and executable by the application server*** (Col. 9, line 1-Col. 11, line 26). As shown in FIG. 7 as a flow chart of access control, an access node process is started and initialized in step 72, and access node parameters concerning other nodes and objects referred to by the node are read from a control file 74. In step 76 the access node is kept in a waiting condition, waiting for incoming user calls 78. If a user call is received, the access node process starts a subprocess in step 80 and sets a timer depending on the control file 82 for the period the subprocess is allowed to exist. In step 84, the subprocess verifies the user, depending on control file 86 containing information about accepted and permitted user identities. The address and the identity of the user is polled in communication 88. If the user is permitted and accepted, the user is logged in to the access node in step 104 and 106 and is allowed access to functions of the access node and objects encapsulated in or referred to by the access node. Thereafter, further access to and communication of data and references contained in the access node is controlled by means of an access filter having certain access control parameters and allowing different users different views of the access node, the access structure and the underlying data. In step 124 of FIG. 8, upon a user command (cmd=dir) all objects are copied to or listed in an object list 128. Thereafter, in step 132 every object in the object list is checked in respect of whether or not the user is permitted access to it. An object access list 134 is thereby used as a check reference. If access right exists for an object, a copy of or a reference to that object is input in a user list, step 136, and is communicated to the user in step 138 by call 140 (Col. 4, lines 39-48; Col. 5, line 63-Col. 6, line 8; Col. 10, line 42-Col. 12, line 11). This indicates ***the application code being responsive to the access criteria associated with the groups of data contained within a version of an object and to predetermined privileges for allowing controlled access to individual groups of data contained within the version of an object by an individual user that was set up to be sent to a user computer system and that may be viewed by a user according to the user's predetermined privileges on the user computer system***. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Thorsen system by including a database for storing object and associated information, an application server controlled by software code in memory, access data application code being responsive to the access criteria associated with the groups of data contained within a version of an object and to predetermined privileges for allowing controlled access to individual groups of data contained within the version of the object by an individual user that was set up to be sent to a user computer an that may be viewed by a user according to the user's predetermined privileges on the user computer system in order to control access to data and providing different views of stored data objects depending on different aspects of the stored objects or different access rights of a user.

Regarding to claim 2, Thorsen teaches all the claimed subject matters as discussed in claim 1 and further discloses: ***the access data application code enables the ability of a user to read the contents of the transferred version of the requested object that was sent by the application server according to access privileges associated with the user*** (Col. 4, lines 39-48; Col. 10, lines 56-60).

Regarding to claim 3, Thorsen teaches all the claimed subject matters as discussed in claim 2, Thorsen further discloses ***the access data application code includes the ability to modify the contents of version of the requested object*** (Col. 10, lines 56-60; Col. 11, lines 22-26).

Regarding to claim 4, Thorsen teaches all the claimed subject matters as discussed in claim 3, Thorsen further discloses ***the ability to modify includes the ability to delete information contained in the version of the requested object*** (Col. 12, lines 32-38).

Regarding to claim 5, Thorsen teaches all the claimed subject matters as discussed in claim 3, Thorsen further discloses ***the ability to modify includes the ability to add data to the version of the requested object*** (Col. 4, lines 39-56; Col. 12, lines 32-38).

Regarding to claim 6, Thorsen teaches all the claimed subject matters as discussed in claim 1, Thorsen does not explicitly teach ***the access to the version of the object is determined by a business relationship to produce products and defined by the host according to the need of information in the product chain, and wherein the transferred***

version of the object is configured to reveal limited information according to a guest user's predetermined access privileges. However, Thorsen teaches: the system to manage and access data of an enterprise with different departments such as finance, production, sales and storage... (Col. 1, lines 14-27) that leads to controlling access as an object of Thorsen system (Col. 3, lines 8-22). This indicates a business relationship that has productions and the need of information in the product chain, also the access right defined by the host. In addition, at step 124 of FIG. 8, upon a user command (cmd=dir) all objects are copied to or listed in an object list 128. Thereafter, in step 132 every object in the object list is checked in respect of whether or not the user is permitted access to it. An object access list 134 is thereby used as a check reference. If access right exists for an object, a copy of or a reference to that object is input in a user list, step 136, and is communicated to the user in step 138 by call 140 (Col. 11, line 34-Col. 12, line 11). This indicates the transferred version of the object is configured to reveal limited information according to a guest user's predetermined access privileges. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Thorsen system to have a business relationship that determine the access to the information in the product chain in order to exchange business data in a proper way.

Regarding to claim 7, Thorsen teaches a method for accessing data in a computer-based data processing comprises: ***storing an object, the object comprising distinguishable groups of data, each group of data having associated access criteria for access to the groups of data*** (FIG. 3B, Col. 8, lines 43-56; Col. 10, line 42-Col. 11, line 26);

storing software code for controlling the operation of the CPU in memory (FIG. 1, Col. 7, lines 5-16). As shown in FIG. 3B, Thorsen discloses a state of the art database 52, storing data in a number of tables 54 and being provided with specific data associations. Depending on a control file, the data associations are rearranged; references or pointers to each of the selected data items of the tables 54 are arranged and stored in a number of data access nodes 56 as ***an application server***. A client 58, for example an application program, communicates with the access nodes 56 of the access structure and a new interface is provided between the old database and the user (Col. 8, lines 42-52). This indicates the step of ***controlling the access to the database using an application server***. As shown in FIG. 8 is a flow chart for implementing an object access control means or an object filter. The object filter is used in conjunction with the access control to protect an object referred to by an access node. In step 124, upon a user command (cmd=dir) all objects are copied to or listed in an object list 128. Thereafter, in step 132 every object in the object list is checked in respect of whether or not the user is permitted access to it. An object access list 134 is thereby used as a check reference. If access right exists for an object, a copy of or a reference to that object is input in a user list, step 136, and is communicated to the user in step 138 by call 140 (Col. 11, line 66-Col. 12, line 11). As shown in FIG. 7 as a flow chart of access control, an access node process is started and initialized in step 72, and access node parameters concerning other nodes and objects referred to by the node are read from a control file 74. In step 76 the access node is kept in a waiting condition, waiting for incoming user calls 78. If a user call is received, the access node process starts a subprocess in step

80 and sets a timer depending on the control file 82 for the period the subprocess is allowed to exist. In step 84, the subprocess verifies the user, depending on control file 86 containing information about accepted and permitted user identities. The address and the identity of the user is polled in communication 88. If the user is permitted and accepted, the user is logged in to the access node in step 104 and 106 and is allowed access to functions of the access node and objects encapsulated in or referred to by the access node. Thereafter, further access to and communication of data and references contained in the access node is controlled by means of an access filter having certain access control parameters and allowing different users different views of the access node, the access structure and the underlying data. In step 124 of FIG. 8, upon a user command (cmd=dir) all objects are copied to or listed in an object list 128. Thereafter, in step 132 every object in the object list is checked in respect of whether or not the user is permitted access to it. An object access list 134 is thereby used as a check reference. If access right exists for an object, a copy of or a reference to that object is input in a user list, step 136, and is communicated to the user in step 138 by call 140 (Col. 4, lines 30-33; Col. 10, lines 47-60; Col. 11, line 34-Col. 12, line 11). This indicates the steps of ***an application server, that is configured to set up a version of an object according to access criteria***; ***transferring a version of an object to a user in the form of a document file having the version of the object and any associated documents request by a user contained therein;*** and ***allowing controlled access to individual groups of data contained within the object by an individual user according to the user's privileges in response to the access criteria associated with the group of data contained within an object and to***

predetermined privileges. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Thorsen method by including the steps of storing an object, controlling the access to the database using an application server, transferring a version of an object to a user and allowing controlled access to individual groups of data contained within the version of the object by an individual user that may be viewed by a user according to the user's predetermined privileges on the user computer system in order to control access to data and providing different views of stored data objects depending on different aspects of the stored objects or different access rights of a user.

Regarding to claim 8, Thorsen teaches all the claimed subject matters as discussed in claim 7, and further discloses the steps of ***receiving an object request by a requestor*** (FIG. 7, Col. 11, lines 41-46); ***verifying the requestor's user privilege access criteria*** (FIG. 7, Col. 11, lines 49-53); and ***transmitting a version of an object configured to reveal information contained with in the version of the object according to the requestor's user privilege access criteria*** (Col. 4, lines 39-48; Col. 10, line 42-Col. 11, line 26; FIG. 7, Col. 11, lines 53-65).

Regarding to claim 9, Thorsen teaches all the claimed subject matters, as discussed in claim 7, and further discloses the steps of ***establishing a version of an object includes loading information into the version of an object into separate groups having separate access privilege criteria*** (Col. 4, lines 39-48; FIG. 2-3A; Col. 8, lines 12-34; Col. 10, line 42-Col. 11, line 26).

Regarding to claim 10, Thorsen teaches all the claimed subject matters as discussed in claim 7, and further discloses the steps of ***establishing privilege access criteria includes identifying the separate groups of information to which the user may access for use in setting up a version of the object to be sent to the user in response to the user request*** (Col. 10, line 42-Col. 11, line 33).

Regarding to claim 11, Thorson teaches all the claimed subject matters as discussed in claim 7, and further discloses the step of ***verifying the requestor's user privilege access criteria includes extracting the requestor's user identification from the object request*** (FIG. 7, Col. 11, lines 34-41); ***verifying the requestor's user identification*** (Col. 11, lines 41-53) and ***identifying the groups of data within the version of the object to which the requestor has access*** (Col. 11, lines 56-65).

Regarding to claim 12, Thorson teaches all the claimed subject matters as discussed in claim 7, and further discloses the step of ***transmitting a redacted version of an object by sending an electronic object to the requestor that contains the groups of information to which the requestor has access to and that excludes groups of information associated with an object to which the requestor does not have access*** (Col. 4, lines 30-33; Col. 10, lines 56-60; Col. 11, line 22-Col. 12, line 11).

(11) Response to Argument

<u>Group I for claims 1, 6-7, 10-12</u>

Appellants argued that:

> *Thorsen does not disclose or suggest accessing database data via an object that is outside the database. The Thorsen disclosure focuses on direct access to the database and fails to suggest*

the type of data access through an object as recited in Claim 1. Thus, Thorsen fails to disclose or suggest the limited access of an object that is outside a database.

Examiner respectfully traverses for the following reasons:

Claim 1 recites *an application server configured to control access to data stored in the database,* and *the application code being responsive to the access criteria associated with the groups of data contained within a version of an object and to predetermined privileges for allowing controlled access to individual groups of data contained within the version of an object.* Thus, claim 1 does not include the limited access of an object that is outside a database as mentioned by the appellants.

FIG. 3B in Thorsen shows a database 52 for storing a number of tables 54 and being provided with specific data associations. Depending on a control file, the data associations are rearranged; references or pointers to each of the selected data items of the tables 54 are arranged and stored in a number of data access nodes, col. 8, lines 35-48; col. 4, lines 30-33. The control file describes the format of the input file and the data items 28 to be produced, a per se known spreadsheet or the like may be used in a further step of the transformation process, col. 8, lines 12-23. As seen, the database and data items in Thorsen perform the claimed *a database for storing a version of an object and association information data, the object comprising distinguishable groups of data.*

Thorsen also discloses that each data item or reference to the data item is provided with a time parameter for indicating time at which the data item is read, stored updated by replacing an old data value with a new one, or by creating a new data item with last data version, col. 4, lines 39-38. Thus, the data items having associated access criteria, such as time parameter. Another word, this teaches the claimed *access criteria associated with the groups of data contained within a version of an object.*

Thorsen further discloses, a client 58 communicates with the access nodes 56 of the access structure and a new interface is provided between the database and the user, col. 8, lines 48-52. The initiation of the access structure may also include establishing communications protocols for communication with and between access nodes 56, col. 10, lines 1-8. Upon a call, the first access node process 40 initiates a second access node process 46, which is a copy of the parent node with all qualities and access parameters. The first access node 40 delegates the communication and access control to second access node processes 46 then acting as session servers. By means of this delegation mechanism, an access node controlling access to certain data items or certain views of the data collection may simultaneously serve a large number of users and user applications, col. 10, lines 20-41. This teaches the claimed *application server configured to control access to data stored in the database.*

If a reading command is received for retrieving the data items, the time variables are set in accordance with specifications given in the reading command, a data item in the data shell is read and if it has the specified time value equal the time variables, this data item is inserted in a result list or result vector. Then, the result list is sent to the user, col. 13, lines 32-47. Thus, the result list having data items serves as the claimed *document file having a representation of an object and associated documents that are stored in the database.*

The data access structures are implemented in the high level programming language C and utilize system calls to the operating system, col. 10, lines 9-19. This teaches the claimed *memory for storing software code for controlling the operation of the application server.*

In order to prevent unauthorized access to data, access control parameters constituting an access filter are comprised in the access nodes. When the access node process is started and initialized, access node parameters concerning other nodes and objects referred to by the node are read from a control file 74. If a user call is received, the access node process starts a subprocess in step 80 and sets a timer depending on the control file 74 for the period the subprocess is allowed to exist. In step 84, the subprocess verifies the user, depending on the control file 74 containing information about accepted and permitted user identities. The access filter has certain access control parameters and allows different users different views of the access node, col. 11, lines 34-65. This teaches the claimed *access data application code stored in the memory and executable by the application server, the application code being responsive access criteria associated with the groups of data contained within a version of an object.*

Table I in Thorsen shows a control file for specific nodes having the sixth field indicates access control or access filter parameters, which represent the relevant access level for that node. For example, "r" means that reading is allowed and that the TCP/IP address to this node may be obtained, col. 10, lines 42-60. Thus, the sixth field teaches the claimed *predetermined privileges for allowing controlled access to individual groups of data contained within the version of an object by an individual user.* The object filter is used in conjunction with the access control to protect an object referred to by an access node. If the user is permitted access to the object, it is input in a user list and is communicated to the user. This step allows the object is sent to the user's system for being retrieved, reviewed, or stored, col. 12, lines 1-11. Thus, the claimed *to be sent to a user computer system and that may be viewed by a user according to the user's predetermined privileges on the user computer system* is taught in Thorsen.

Group II for claims 2-5 and 8

Appellants argued that:

> *Thorsen does not disclose providing an object based on access privileges associated with the user. Since Thorsen fails to disclose the use of an object as recited in Claim 2, Thorsen makes no reference or suggestion to an object that is based on a user's access privileges. Thus, Thorsen fails to disclose or suggest access data application code that allows a user to read the contents of an object according to access privileges associated with the user.*

As discussed in the above Group I, the sixth field in Thorsen determines the *access privileges associated with the user*. Fig. 7 illustrates a subprocess for verifying the user. If the user is denied access, the subprocess is terminated and a message is sent to the user. If the user is permitted and accepted, the user is logged, and allowed access to functions of the access node, col. 11, lines 49-61. The object filter is used in conjunction with the access control to protect an object referred to by an access node. If access right exists for an object, a copy of or a reference to that object is input in a user list, and is communicated to the user, col. 12, lines 1-11. In either situation of denying or granting access, via communication, the user is received a message or content of an object, which is reviewed by the user. Thus, Thorsen teaches the claimed *enables the ability of a user to read the contents of the transferred version of the requested object.*

Group III for claim 9

Appellants argued that:

> *Thorsen does not disclose the use of an object containing different groups of information having different access criteria. Since Thorsen fails to disclose the use of this type of object, there is no suggestion in Thorsen to include an object with groups of information having different access criteria. Thorsen describes a system that provides direct access to a database rather than a system using objects containing groups of information with different access criteria, as claimed in Claim 9. Thus, Thorsen fails to disclose or suggest loading information into a version of an object in separate groups having separate access privilege criteria.*

Depending on a control file, the data associations are rearranged; references or pointers to each of the selected data items or objects of the tables 54 are arranged and stored in a number of data access nodes, col. 8, lines 35-48. Figure 3B shows each element in the table 54 loads different access nodes 56. These nodes have different fields for indicating differences levels of access, such as data, type, identity, access right, col. 10, lines 42-60. The access nodes include access control parameters constituting an access filter, which is arranged to let different interested parties or clients have their specific view of the stored object, col. 11, lines 27-33. Thus, Thorsen teaches the claimed *loading information into a version of an object in separate groups having separate access privilege criteria*.

Group IV for claims 13-15

Appellants argued that:

> *Thorsen does not disclose transmitting a redacted version of an object that restricts information. Although Thorsen discloses a system that provides direct access to a database, the reference fails*

to disclose all elements of Claims 13, 14 and 15. Accordingly, Claims 13, 14, and 15 are not anticipated by Thorsen because Thorsen fails to disclose transmitting a redacted version of an object that restricts information according to the requestor's user privilege access criteria.

Regarding to claim 13, Thorsen teaches a computer program product for use with a data processing and storage system and for obtaining a view of a database or a subset of a database, Fig. 1, col. 7, lines 5-28. FIG. 3B in Thorsen shows a database 52 for storing a number of tables 54 and being provided with specific data associations. Depending on a control file, the data associations are rearranged; references or pointers to each of the selected data items of the tables 54 are arranged and stored in a number of data access nodes, col. 8, lines 35-48; col. 4, lines 30-33. The control file describes the format of the input file and the data items 28 to be produced, a per se known spreadsheet or the like may be used in a further step of the transformation process, col. 8, lines 12-23. As seen, the data items in Thorsen perform the claimed *computer readable code means for establishing an object in a storage location.*

When the access node process is started and initialized, access node parameters concerning other nodes and objects referred to by the node are read from a control file 74. If a user call is received, the access node process starts a subprocess in step 80 and sets a timer depending on the control file 74 for the period the subprocess is allowed to exist. In step 84, the subprocess verifies the user, depending on the control file 74 containing information about accepted and permitted user identities, col. 11, lines 34-52. Thus, the setting of a time and the verifying user process teach the claimed *computer readable code for identifying a user to have limited access to information associated with the object.*

Table I in Thorsen shows a control file for specific nodes having the sixth field indicates access control or access filter parameters, which represent the relevant access level for that node, col. 10, lines 42-60. Access control parameters include read and write operation, col. 15, lines 18-21. Thus, the sixth field teaches the claimed *computer readable code means for establishing privilege access criteria that define the scope of access of a version of the object for the user.*

An object filter is used in conjunction with the access control to protect an object referred to by an access node. In step 124 of Fig. 8, upon a user command (cmd=dir) all objects are copied to or listed in an object list 128, col. 11, line 66-col. 12, line 5. The user command teaches the claimed *computer readable code for receiving an object request by a requestor.*

In order to enable a mechanism allowing certain users to see references to other access nodes and other users to see only the data or object referred, the access node comprises one access rights list for every object, col. 12, lines 12-16. Every object in the object list is checked in respect of whether or not the user is permitted access to it. An object access list 134 is used as a check reference. If access right exists for an object, a copy of or a reference to that object is input in a user list, col. 12, lines 5-10. This teaches the claimed *computer readable code means for verifying the requestor's user privilege access criteria.*

If a reading command is received for retrieving the data items, the time variables are set in accordance with specifications given in the reading command, a data item in the data shell is read and if it has the specified time value equal the time variables, this data item is inserted in a result list or result vector. Then, the result list is sent to the user, col. 13, lines 32-47. All data is provided with a time parameter stored in connection to each data item. This feature is due to the recognition of the fact that data in different context mainly differ in the frequency of changes. Static data has a value with a changing frequency of zero changes per time unit, whereas variable data may vary discretely or continuously at any rate, col. 12, lines 18-30. Hence, a means is provided which allows a database administrator to erase data or selected parts of the data, i.e., redact. All data, the last data, or any selected data from a certain point of time is read by authorized users, col. 12, lines 32-38. At certain selected period of time, the result list sent to the user includes the data has been edited (erase, select). Thus, the claimed *computer readable code means for transmitting a version of the requested object in the form of a redacted document that masks information according to the requestor's user privilege access criteria is anticipated by Thorsen.*

<u>Group V for claim 16</u>

Appellants argued that:

> *Thorsen does not disclose the use of an object that is set up based on a user's access privileges. Thus, Thorsen fails to disclose all elements of Claim 16. Accordingly, Claim 16 is not anticipated by Thorsen because Thorsen fails to disclose transmitting a version of an object that was set up according to a user's privilege access criteria in the form of a document file that includes a version of the requested object and a version of associated documents.*

Fig. 7 illustrates a subprocess for verifying the user. If the user is denied access, the subprocess is terminated and a message is sent to the user. If the user is permitted and accepted, the user is logged, and allowed access to functions of the access node, col. 11, lines 49-61. This teaches the claimed *verifying the requestor's user privilege access criteria.*

The object filter is used in conjunction with the access control to protect an object referred to by an access node. If access right exists for an object, a copy of or a reference to that object is input in a user list, and is communicated to the user, col. 12, lines 1-11. In either situation of denying or granting access, via communication, the user is received a message or content of an object. Thorsen also teaches that user may subscribe, i.e., set up, an updating subscription means for automatically received every updating of information or event related, to an object or object attribute. The updating information is associated or added to the control file of the access note, and is sent to the subscriber, i.e., user. Thus, Thorsen teaches the claimed *transmitting a version of an object that was set up according to a user's privilege access criteria in the form of a document file that includes a version of the requested object and a version of associated documents.*

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Examiner: Hung Pham
June 9, 2003

KIM VU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Conferee 1: Kim Vu, SPE, AU 2172 K

Conferee 2: Hossain Alam, SPE, AU 2155 HTA

HOSAIN T. ALAM
PRIMARY EXAMINER